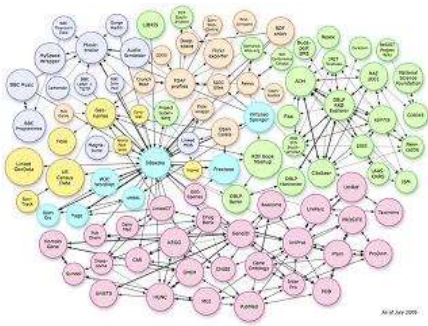
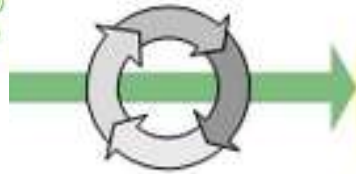


Course Offering Determination in an Extended Ecosystem *(supervised by Prof Fuhua Lin)*



**Learning
Material**



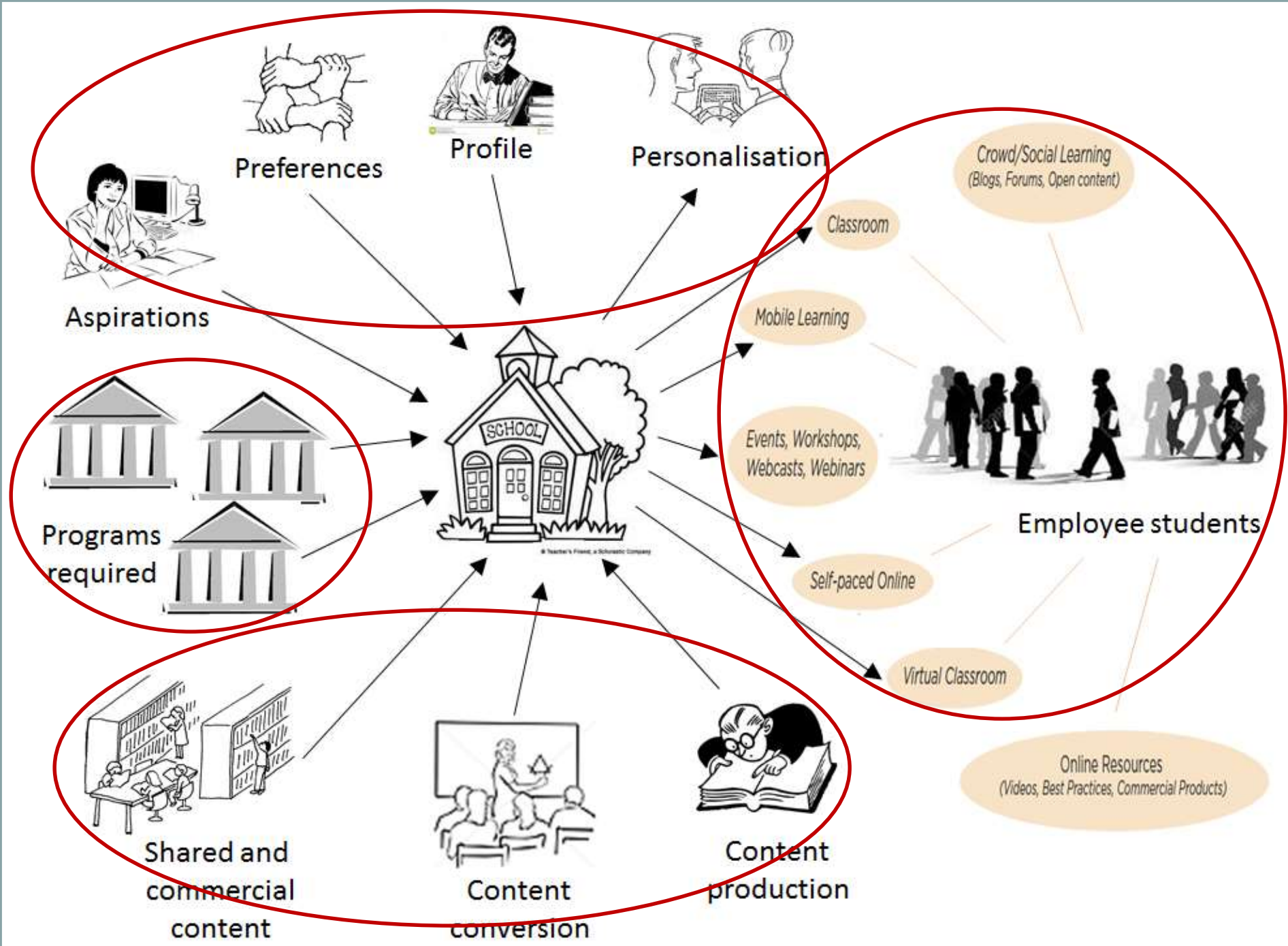
Metadata

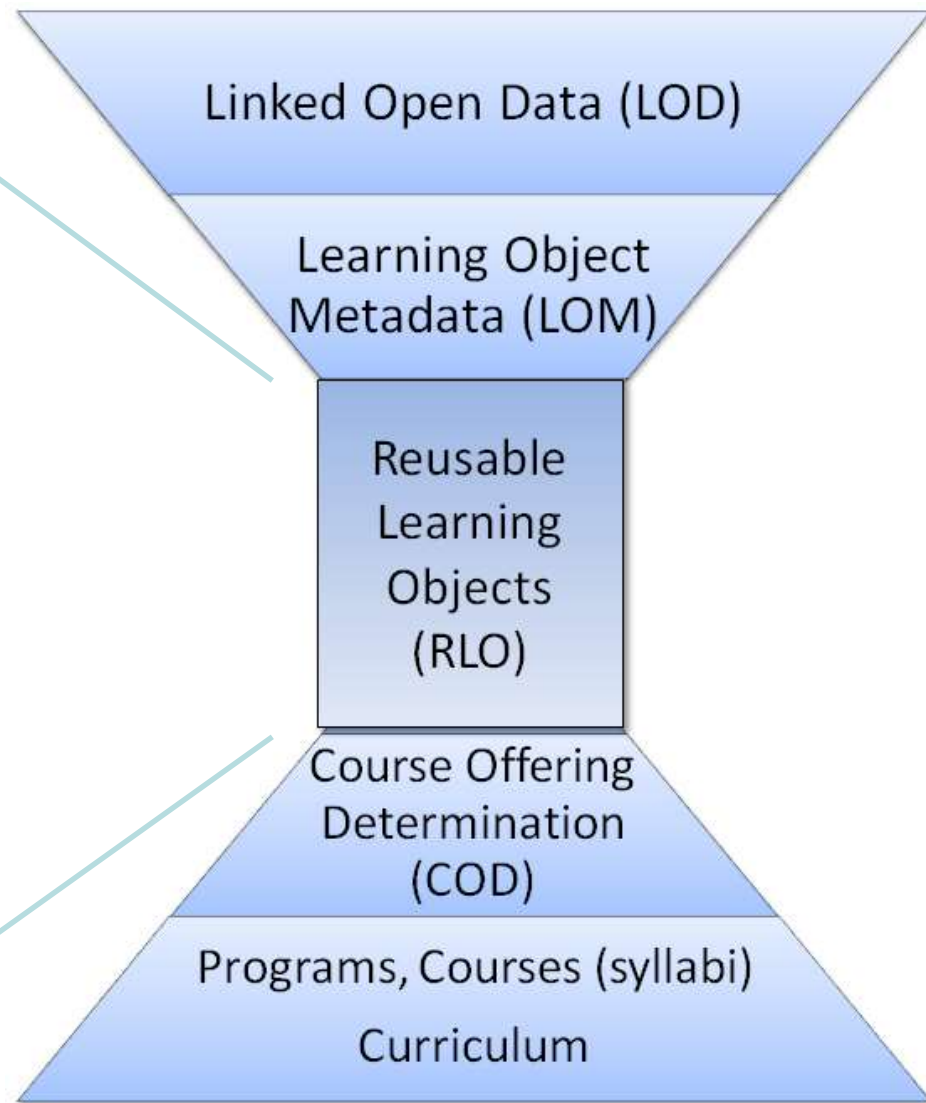
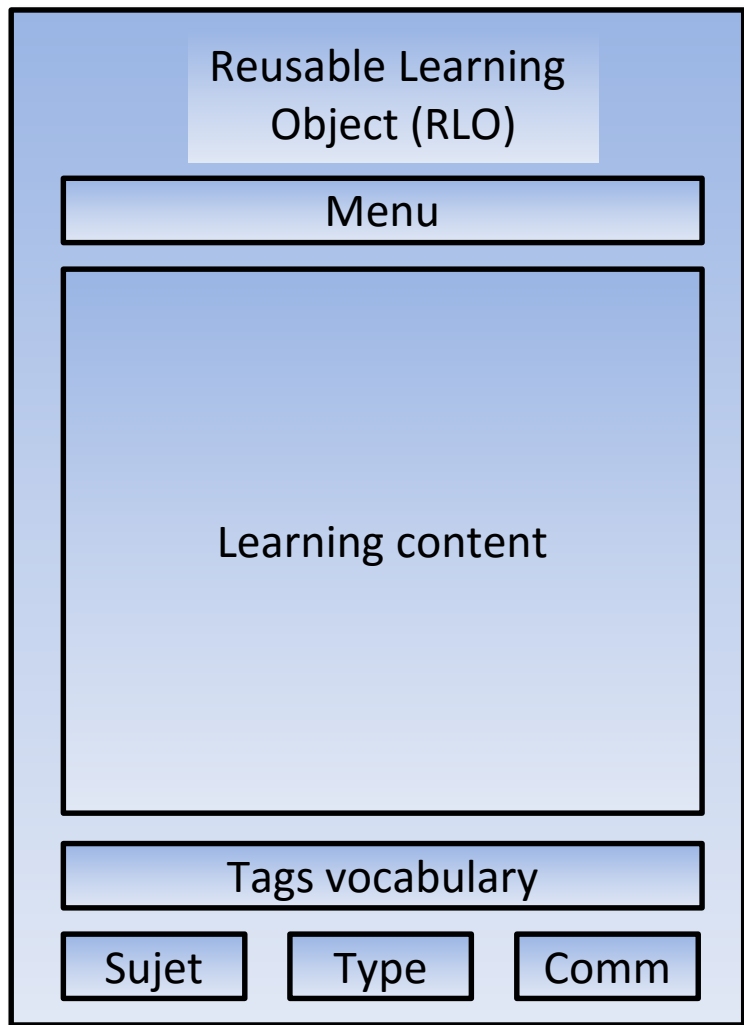
**Pedagogical
ONTOLOGY**

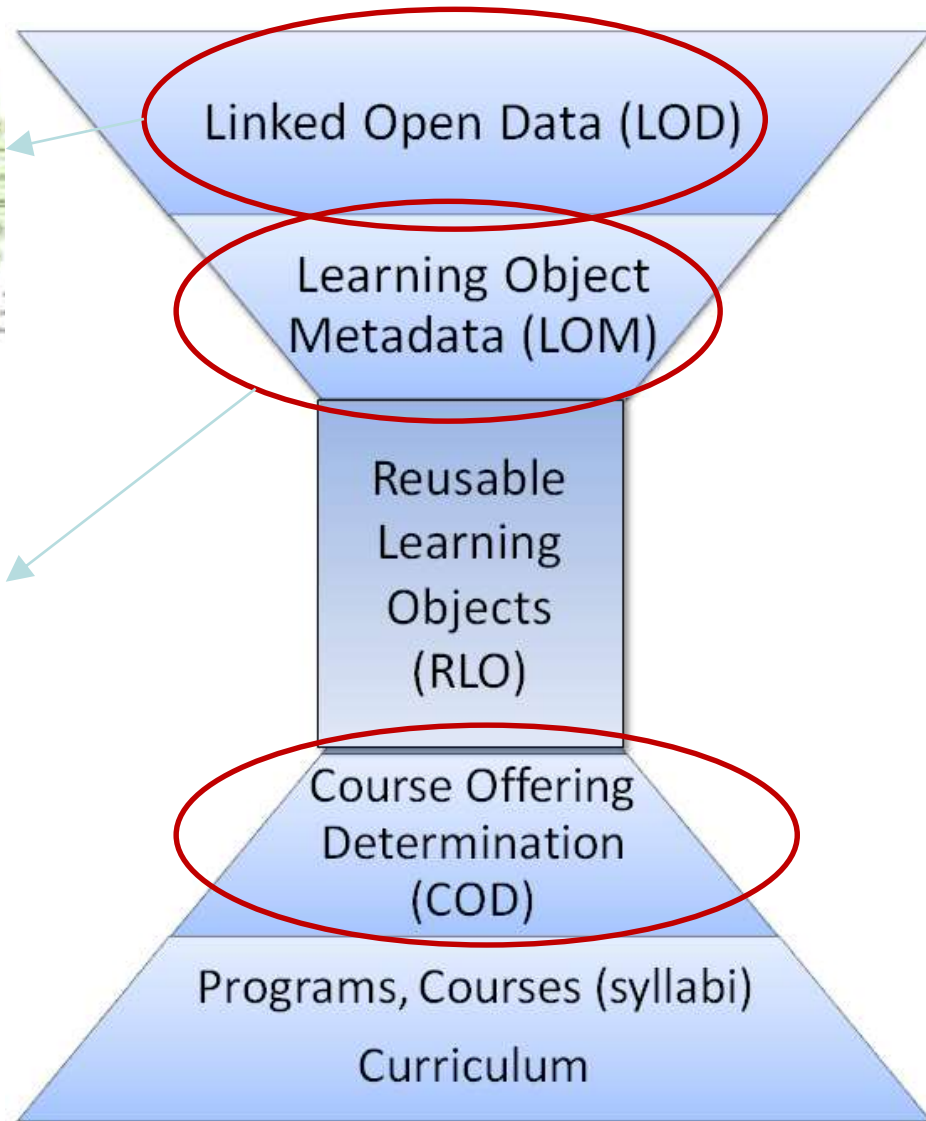
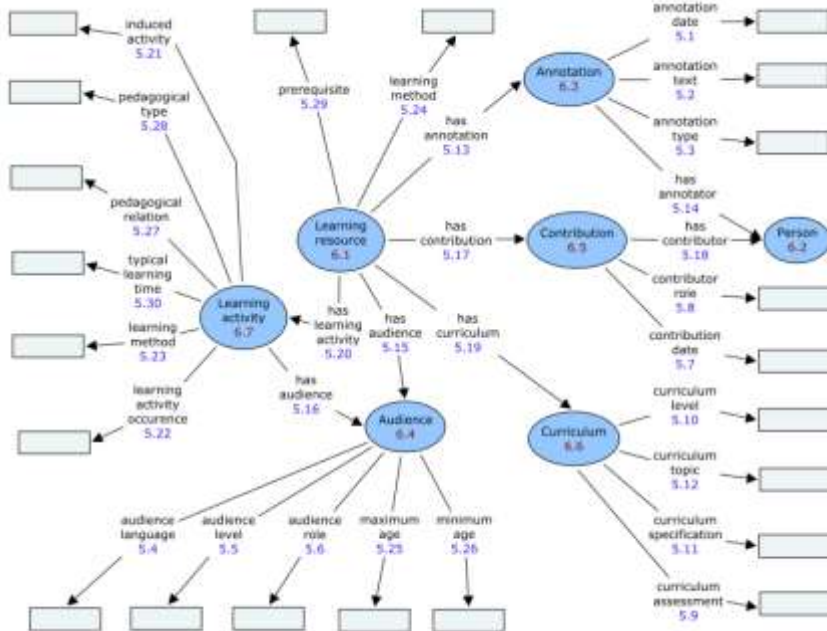
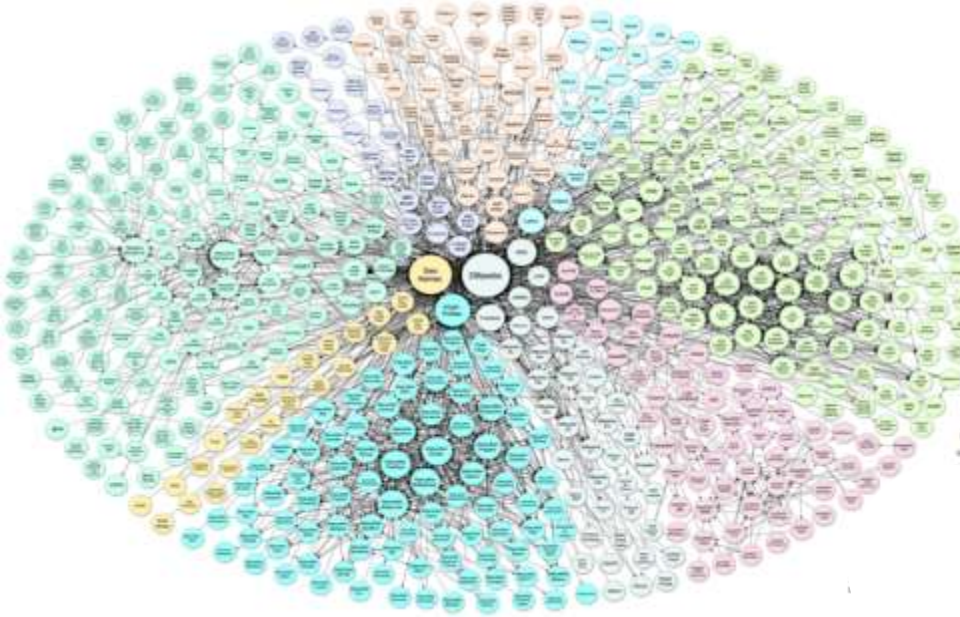
Semantic Technologies
Reusable Learning Objects

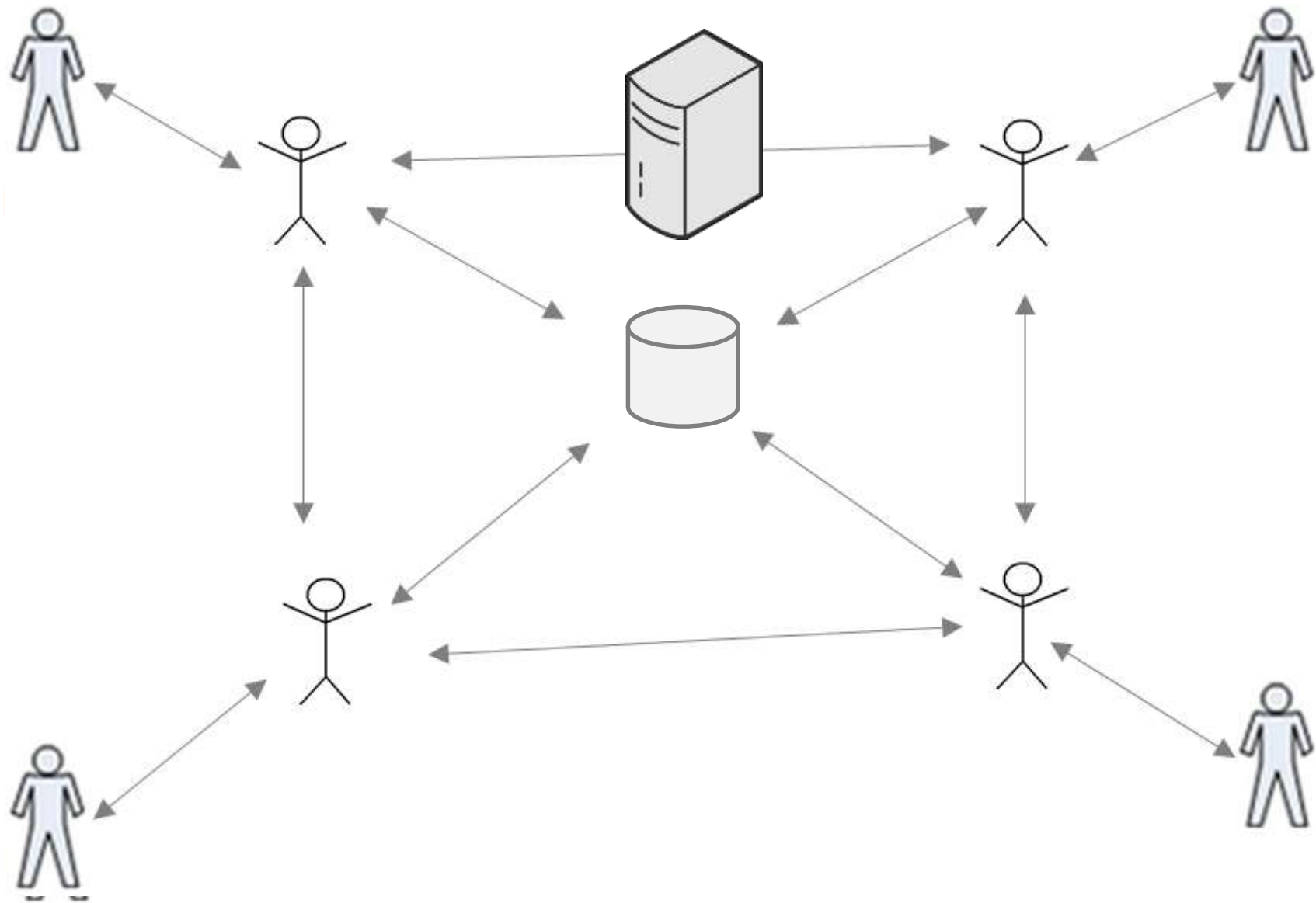
Multi-Agent System
Course Offering Determination

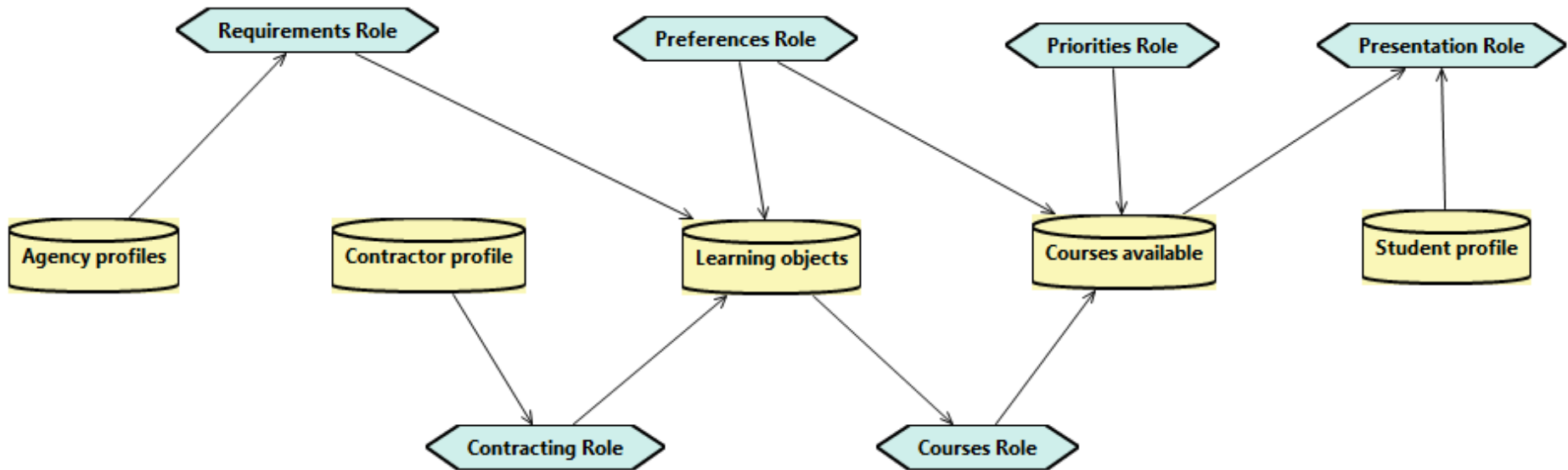
Software Agents for Personalised Learning

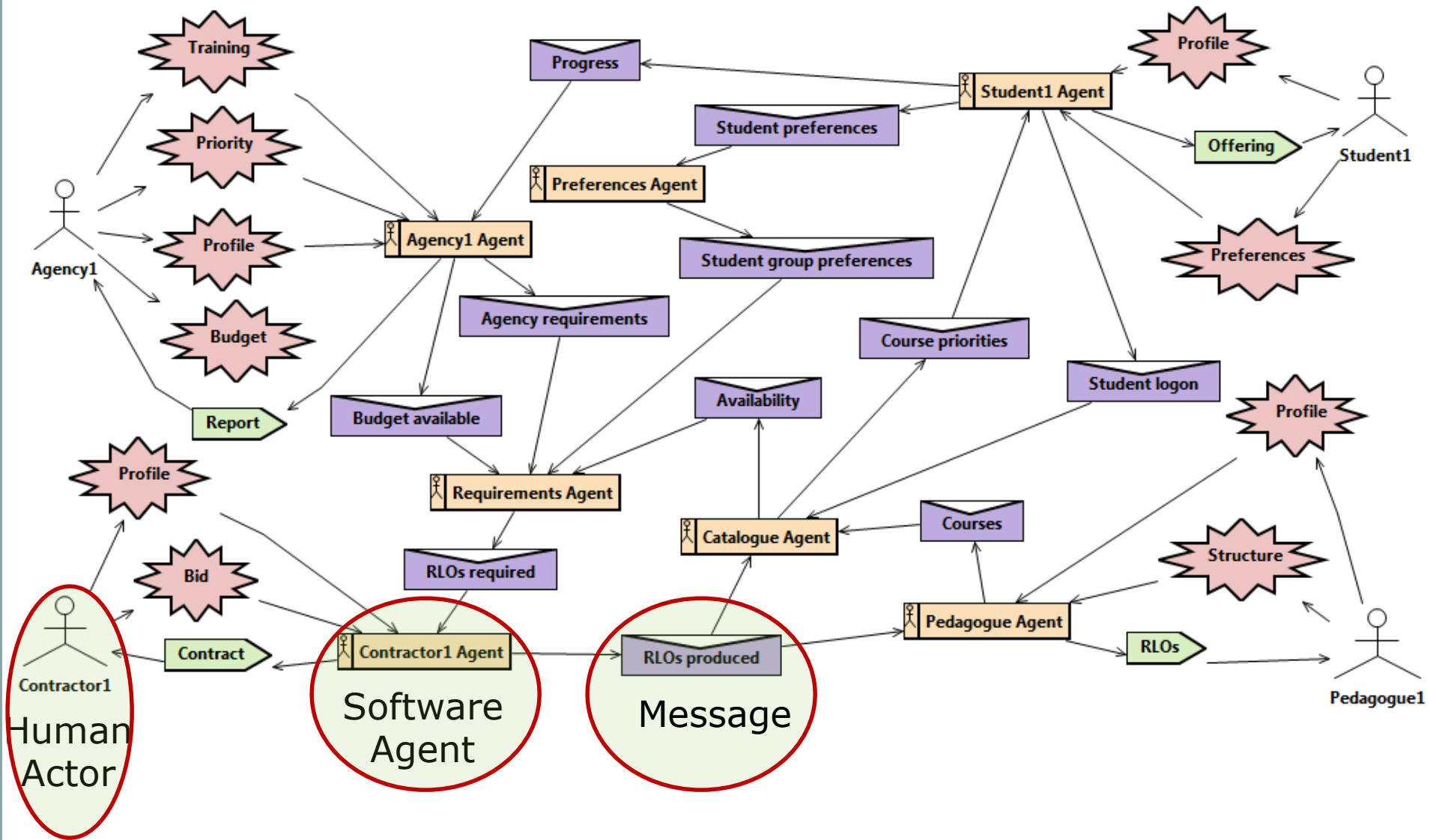


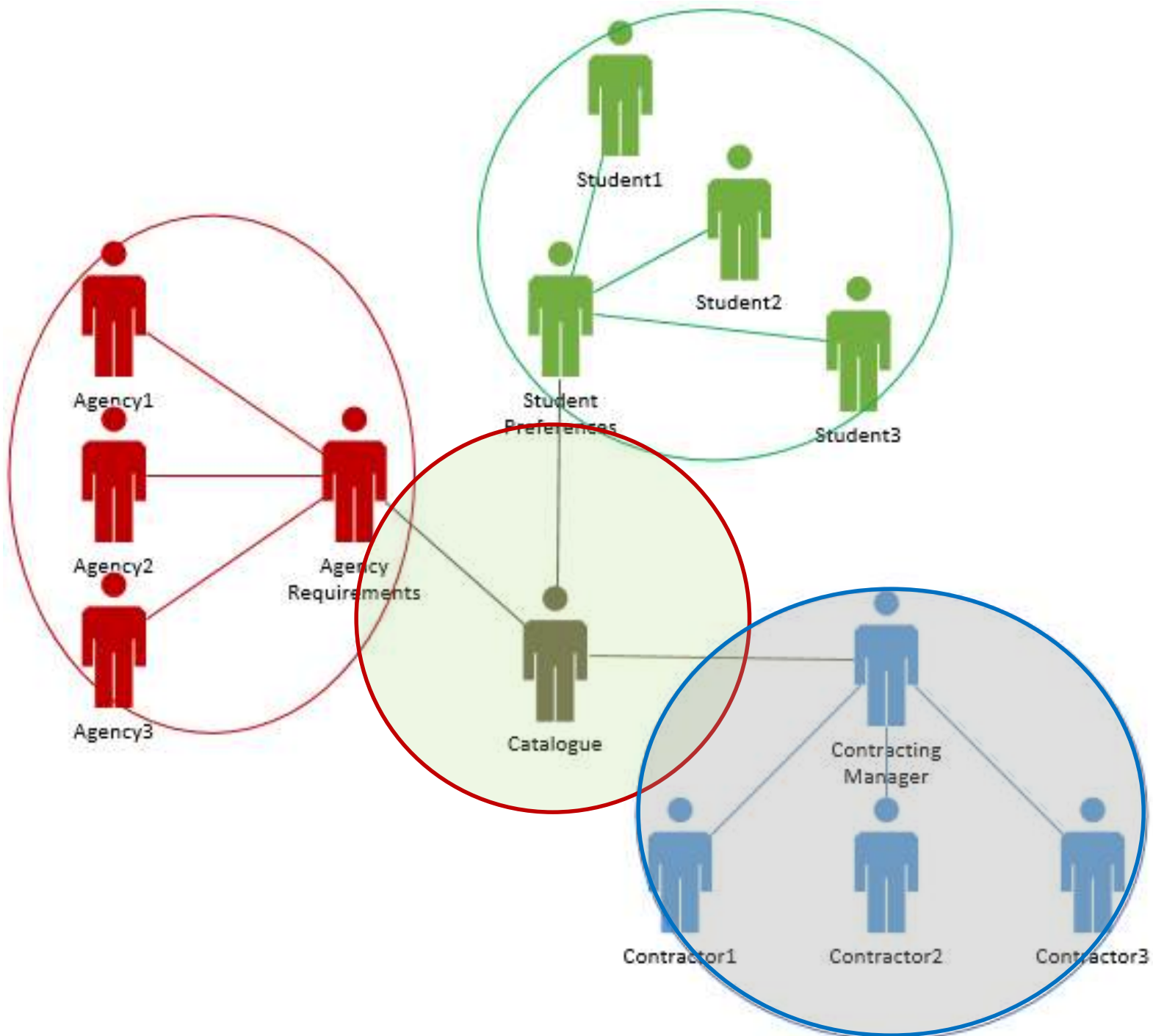












Agency_Requirements

Student_Preferences

Course_Offering

Catalogue_agent

Contracting_agent

Contractor_3

Contractor_2

Contractor_1

RLO_produced

RLO_specification

Budget_available

call for proposal

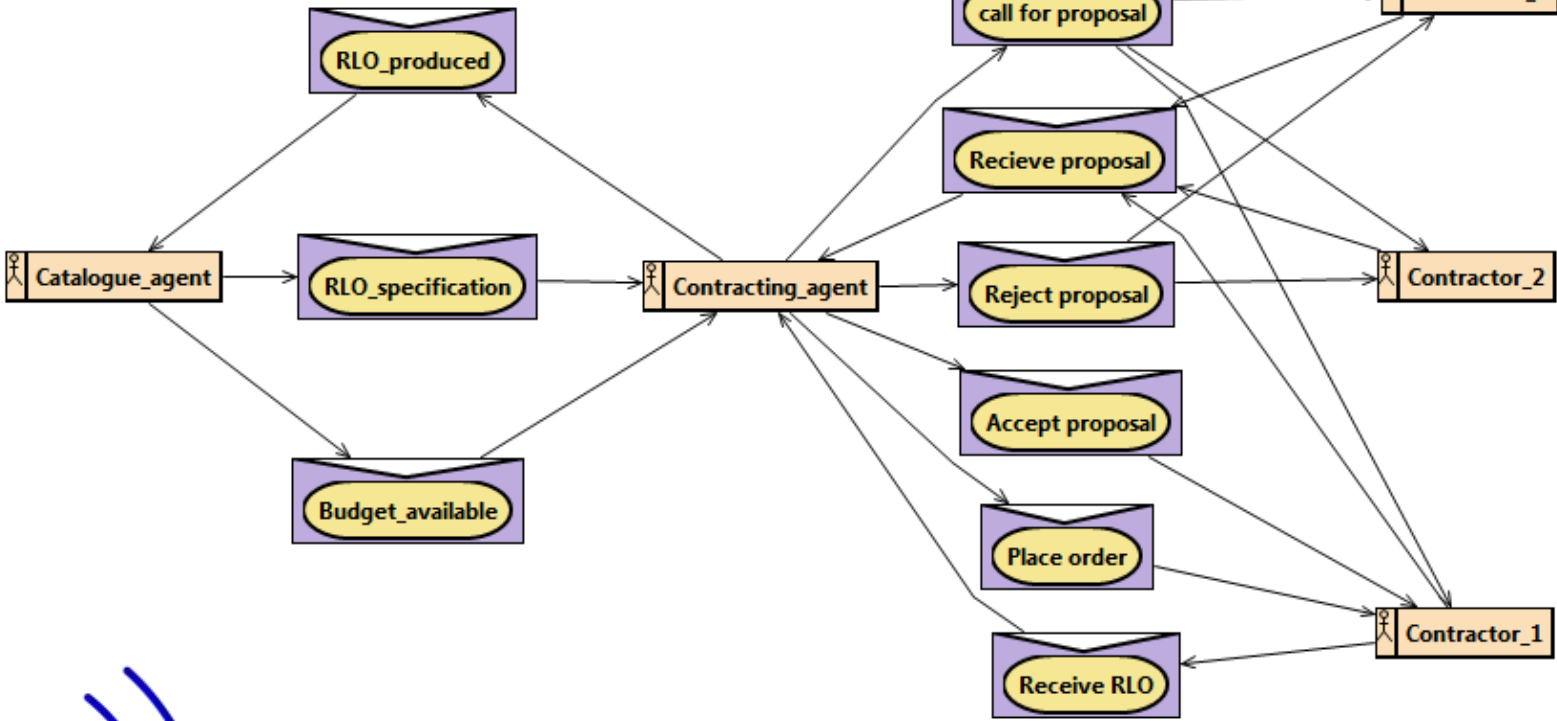
Recieve proposal

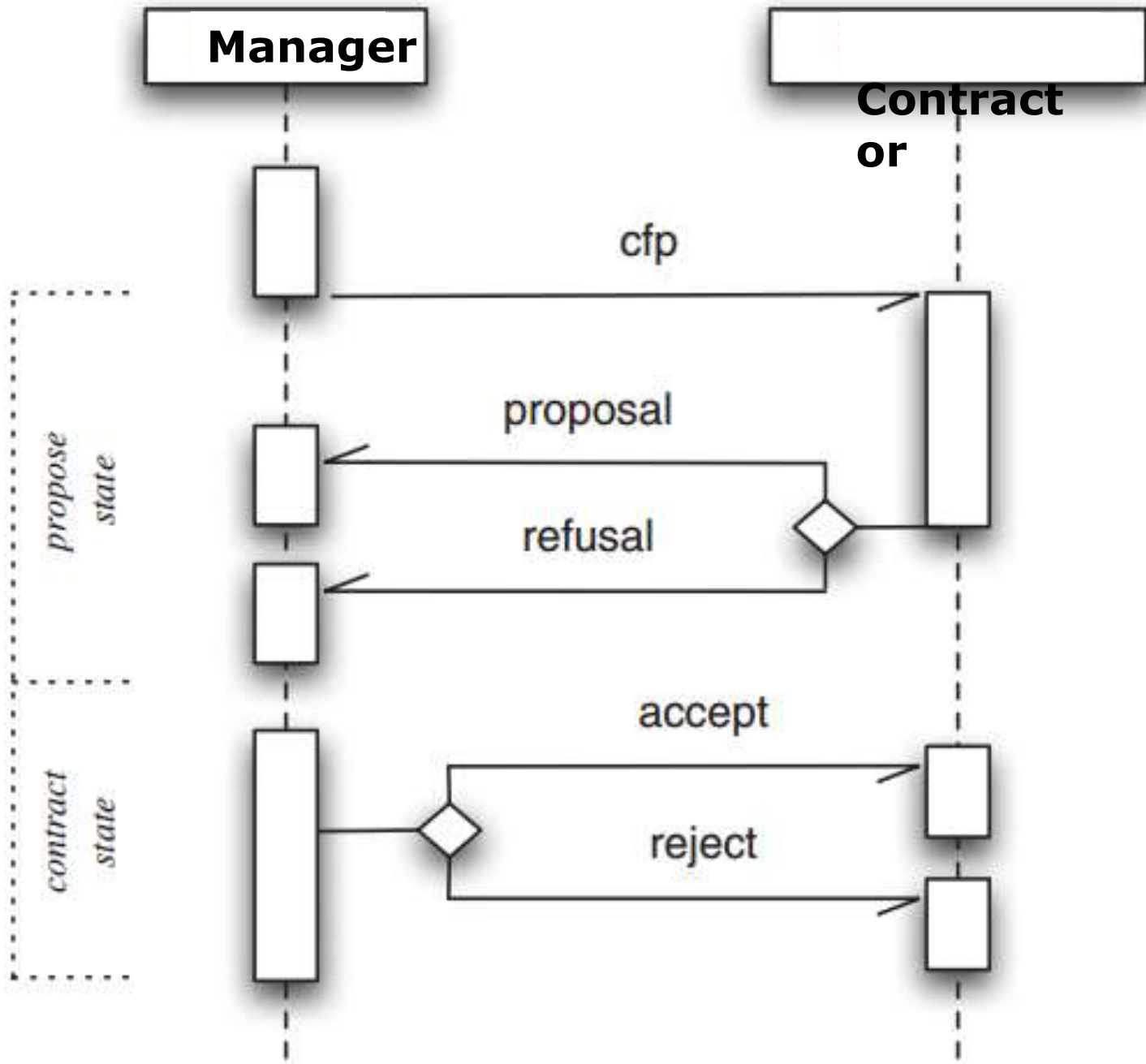
Reject proposal

Accept proposal

Place order

Receive RLO





Actions About

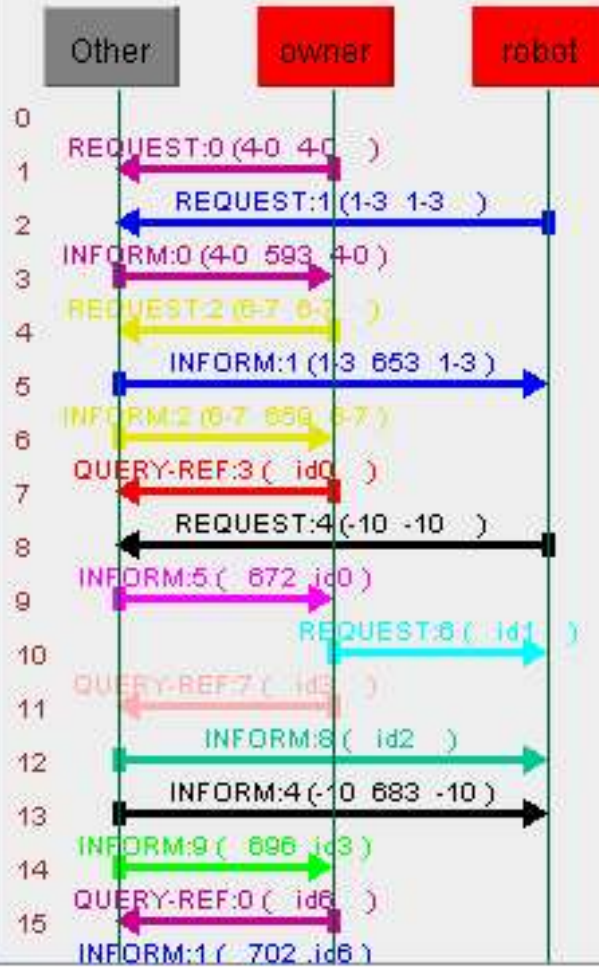


Platforms

isPlatform

Main-Container

- RMA@192.168.0.100:1099/JADE
- Sniffer-on-Main-Container@192.168.0.100
- Sniffer@192.168.0.100:1099/JADE
- ams@192.168.0.100:1099/JADE
- df@192.168.0.100:1099/JADE
- j_environment@192.168.0.100:1099/JADE
- owner@192.168.0.100:1099/JADE
- robot@192.168.0.100:1099/JADE
- supermarket1@192.168.0.100:1099/JADE
- supermarket2@192.168.0.100:1099/JADE
- supermarket3@192.168.0.100:1099/JADE



COD2.mas2j

◇ Infrastructure: Centr

+ constructor (C:\U

+ agency (C:\Users

+ student (C:\Users

+ contractor (C:\Us

+ preferences (C:\U

+ requirements (C:

+ catalogue (C:\Us

+ pedagogue (C:\U

◇ COD2.mas2j

📄 constructor.asl

```
1 MAS cod2 {
2     infrastructure: Centralised
3     agents:
4         constructor;
5         agency #3;
6         student #3;
7         contractor #3;
8         // preferences [verbose = 2];
9         preferences;
10        requirements;
11        catalogue;
12        pedagogue;
13    }
14
15    // the constructor builds the initial belief base and plans for other
16    // agencies are employers proscribing courses needed
17    // students are employees who express preferences via preferences agent
18    // contractors produce RLOs
19    // manages group individual students preferences to compete for RLOs
20    // contracts developments depending on
21    //     students preferences,
22    //     agency requirements, and
23    //     current inventory of RLOs
24    // the catalogue agent provides info to requirements and students */
25    // the pedagogue agent builds courses from RLOs
26
```

```

1 !beliefs.
2 !plans.
3 !goals.
4
5 +!beliefs <-
6     .send(contractor1, tell, rate(101));
7     .send(contractor2, tell, rate(102));
8     .send(contractor3, tell, rate(103)).
9
10 +!plans <-
11     .send(requirements, tellHow, "+!require(X) <- .print(X,[required]); .send(contractor1, achieve, require(X)).");
12     .send(contractor1, tellHow, "+!require(X) <- .print(X,[produced]); .send(pedagogue, tell, rlo(X)).");
13     .send(pedagogue, tellHow, "+rlo(X) <- .print(X,[included_in_curriculum]); .send(catalogue, tell, course(X)).");
14     .send(catalogue, tellHow, "+course(X) <- .print(X,[added_to_catalogue]).");
15     .send(student1, tellHow, "+course(X) <- .print(X).");
16     .send(student1, tellHow, "+!request <- .send(catalogue, askAll, course(X)); .print(courses_available).").
17
18 +!goals <-
19     .send(requirements, achieve, require(rlo101));
20     .send(requirements, achieve, require(rlo102));
21     .send(requirements, achieve, require(rlo103));
22     .send(requirements, achieve, require(rlo201));
23     .send(requirements, achieve, require(rlo202));
24     .send(requirements, achieve, require(rlo203));
25     .send(requirements, achieve, require(rlo301));
26     .send(requirements, achieve, require(rlo302));
27     .send(requirements, achieve, require(rlo303));
28     .wait(100);
29     .send(student1, achieve, request):

```

Jason Http Server running on http://192.168.0.102:3273

```
[requirements] rlo101[required]
[requirements] rlo102[required]
[requirements] rlo103[required]
[requirements] rlo201[required]
[contractor1] rlo101[produced]
[contractor1] rlo102[produced]
[contractor1] rlo103[produced]
[requirements] rlo202[required]
[pedagogue] rlo101[included_in_curriculum]
[pedagogue] rlo102[included_in_curriculum]
[catalogue] rlo101[added_to_catalogue]
[pedagogue] rlo103[included_in_curriculum]
[catalogue] rlo102[added_to_catalogue]
[catalogue] rlo103[added_to_catalogue]
[requirements] rlo203[required]
[requirements] rlo301[required]
[requirements] rlo302[required]
[requirements] rlo3o3[required]
[contractor1] rlo201[produced]
[contractor1] rlo202[produced]
[pedagogue] rlo201[included_in_curriculum]
[contractor1] rlo203[produced]
[catalogue] rlo201[added_to_catalogue]
[pedagogue] rlo202[included_in_curriculum]
[contractor1] rlo301[produced]
[contractor1] rlo302[produced]
[pedagogue] rlo203[included_in_curriculum]
[catalogue] rlo202[added_to_catalogue]
[contractor1] rlo3o3[produced]
[pedagogue] rlo301[included_in_curriculum]
[pedagogue] rlo302[included_in_curriculum]
[pedagogue] rlo3o3[included_in_curriculum]
[catalogue] rlo203[added_to_catalogue]
[catalogue] rlo301[added_to_catalogue]
[catalogue] rlo302[added_to_catalogue]
[catalogue] rlo3o3[added_to_catalogue]
[student1] courses_available
[student1] rlo3o3
[student1] rlo302
[student1] rlo301
[student1] rlo203
[student1] rlo202
```

Questions?

1. [Overview \(end-to-end\)](#)
2. [eSchool Ecosystem](#)
3. [Reusable Learning Obj](#)
4. [RLO in Context](#)
5. [S/W Agents for Humans](#)
6. [Data Coupling view](#)
7. [Messages Overview](#)
8. [Agent Organisation](#)
9. [Segregated Comm's](#)
10. [Contract Net Protocol](#)
11. [Communication Sniffer](#)
12. [Agents in Jason](#)
13. [Programming Agents](#)
14. [Independent Execution](#)